

**WEBSTATION: CONFIGURABLE WEB-BASED WORKSTATION
FOR REASON DRIVEN DATA ANALYSIS**

Inventors

Craig Nies, Jean de Traversay, Arati S. Deo, Anu K. Pathria, and L. Steven Biafore

Prepared by:

Robert R. Sachs

Registration No. 42,120

Fenwick & West LLP

Two Palo Alto Square

Palo Alto, CA 94306

Express Mail No.: EL550795015US

Filed: September 27, 2000

WEBSTATION: CONFIGURABLE WEB-BASED WORKSTATION FOR REASON DRIVEN DATA ANALYSIS

Inventors

Craig Nies, Jean de Traversay, Arati S. Deo, Anu K. Pathria, and L. Steven Biafore

5 Cross Reference to Related Application

This application claims priority under 35 U.S.C. §119(e) from provisional application serial number 60/157,226 filed September 30, 1999, which is incorporated by reference herein in its entirety.

10 Background

Field of the Invention

The present invention relates to workstations and user interfaces for data mining generally, and more particularly, for analyzing and manipulating the results of predictive models.

15 Background of the Invention

In data mining applications generally, and in applications for analyzing the results of predictive models, it is helpful for users to be able to easily traverse through very complex sets of data describe that describe the relationships between different entities. Conventional data mining tools typically allow for providing complex queries on the data set, but still require the user to sufficiently understand the problem and the data so as to know which queries to construct. This makes it relatively more difficult for the user to explore the data in order to find patterns or relationships of interest, without knowing such relationships ahead of time.

It is further desirable to provide a development platform for a database mining application that allows for rapid development and customization of such applications. Many existing database mining applications are hardcoded, and require customized development by experienced programmers.

Summary of the Invention

The present invention overcomes the limitations of existing database mining platforms by providing methods for viewing the results of predictive models with report trees. The predictive model scores entities and ranks them by their score. The report tree 5 includes a number of hyperlinked reports, including a summary level report that summarizes data that contributes to a reason that the entity was included in the scored output. The hyperlinking allow a user to quickly navigate through a complex collection of data, to identify, for example, suspicious or fraudulent activity by an entity.

For example, a predictive model may score healthcare providers for the likelihood 10 of fraud based on their reimbursement claims for healthcare services to patients (reimbursements may be requested after the service is provided, at the time is being provided, or prior to the service being provided). The predictive model provides a ranked order listing of providers, and a reason that each provider is included. The reason may be based on some significant aberration in the provider's claims. The report tree allows a 15 user to explore this difference by exploring complex data relationships and underlying individual transactions (e.g. claims), and by statistically comparing the provider's activities with activities of the provider's peers. One aspect of the invention allows the user to dynamically select different peer groups for comparing and contrasting an entity's behavior.

20 A separate feature of the invention is the ability to create web sites on the fly which contain selected reports from a report tree. A user, while investigating an entity via the report tree, and selectively include different reports into a casebook which forms a web site that others can access to very quickly see the reports of interest.

Yet another separate feature of the invention is a frame based user interface layout 25 for web pages that includes a main menu frame, a display frame, a context menu frame, and a navigation frame.

The features and advantages described in this summary and the following detailed description are not all-inclusive, and particularly, many additional features and advantages will be apparent to one of ordinary skill in the art in view of the drawings, 30 specification, and claims hereof. Moreover, it should be noted that the language used in

the specification has been principally selected for readability and instructional purposes, and may not have been selected to delineate or circumscribe the inventive subject matter, resort to the claims being necessary to determine such inventive subject matter.

5 Brief Description of the Drawings

Fig. 1 is an illustration of an exemplary predictive solution system providing a WebStation interface.

Fig. 2 is an illustration of the development and execution phases of a predictive model system.

10 Fig. 3 is an illustration of the deployment environment for a WebStation.

Fig. 4 is an example of a report tree.

Fig. 5 is a schematic illustration of the report frame layout for the user interface of the WebStation.

Fig. 6a is an illustration of a sample 2-By Peer chart layout

15 Fig. 6b is an illustration of a distribution plot.

Fig. 7 is a logon screen.

Fig. 8 is a user's home page in the WebStation.

Fig. 9 is a suspect list page.

Fig. 10 is a provider summary page.

20 Fig. 11 is a graphical high-level summary page.

Fig. 12 is a patient list page.

Fig. 13 is a patient specific claim data page.

Fig. 14 shows the use of the filter function in the context of the patient specific claim data page.

25 Fig. 15 shows the use of the sort function in the context of the patient specific claim data page.

Fig. 16 illustrates the layout function in the context of the patient specific claim data page.

Fig. 17 illustrates the Procedure and/or Diagnosis Groups report.

30 Fig. 18 illustrates the Flip Client Summaries report.

5

Fig. 19 illustrates the Provider Chart report.

Fig. 20 illustrates the Client Age Breakdown report.

Fig. 21 illustrates the Monthly Activity report.

Fig. 22 illustrates the Client Consecutive Visits report.

Fig. 23 illustrates the Group Participation report.

Fig. 24 illustrates the Dollars Per Claim report.

Fig. 25 illustrates the Per Day Activity report.

Fig. 26 illustrates the Per Client Volume report.

Fig. 27 illustrates the Multiple Physicians Per Day report.

10 Fig. 28 illustrates the relationship of the various case management screens.

Fig. 29 illustrates a sample case list.

Fig. 30 illustrates a sample new case message screen.

Fig. 31 illustrates a sample case summary screen.

Fig. 32 illustrates sample find target screen.

15 Fig. 33 illustrates a sample action screen.

Fig. 34 illustrates a sample case log screen.

Fig. 35 illustrates a sample search screen.

Fig. 36 illustrates a sample case list screen.

20 The figures depict a preferred embodiment of the present invention for purposes of illustration only. One skilled in the art will readily recognize from the following discussion that alternative embodiments of the structures and methods illustrated herein may be employed without departing from the principles of the invention described herein.

25

Detailed Description

I. Overview of WebStation System Architecture

II. A Deployed Predictive Solution Using a Webstation

III. Overview of WebStation Features

Reason-Driven Report Tree

WebStation Frame Layout

WebStation Integration with Case-Management

IV. Spyder Analysis Workstation: A Quick Introduction

A Browser-Based Approach to Viewing and Analyzing Detection Results

A Sample WebStation Session

User Log-on

Primary Detection Page

The Suspect List is the “Root” of Spyder’s Report Tree

Viewing Spyder’s Standard Summary Reports

V. Report Tree Example

VI. WebStation Functional Description

Viewing Report Tree Supporting Data

Navigation Frame content:

Function Frame content:

Data Frame content:

Produce Hardcopy and Electronic Reports

Provide Security and Access Controls

Manage Cases

Additional WebStation Features

Optional Features

	Web CaseBook
	Enhanced Graphics
	Graphing Function
	File Export
5	<i>VII. Case Management Module Software</i>
	Functional Overview
	Manage Cases
	Assign Cases
	Generate Management Reports
10	Case Management and Report Navigation
	Web CaseBook
	<i>VIII. WebStation Configuration File</i>
	Preferred Syntax of a Configuration File
	Pages
15	Title
	SQL
	Coloring
	Column
	Shown / Order
20	Example Page Configuration

I. Overview of WebStation System Architecture

The Analysis WebStation (WebStation) provides a system, method, and software product with a user interface that enables end-users to view, understand, analyze and act upon the results of predictive models. Though the WebStation may also be used with
25

other application areas using various types of transactional data, in this disclosure, an exemplary embodiment for predictive solutions is discussed.

Referring now to Fig. 1, there is shown an overview of the system architecture of a system 100 for implementing the WebStation 150. Each element of the system of Fig. 1 can run on a different hardware platform, or may be implemented by consolidating the elements on a more limited number of platforms. The Data Source 110 refers to the raw data upon which the predictive decision will be based. This source often consists of transaction data and supporting data such as demographics or master-file data describing individual “entities.” In one embodiment, such as for analysis of healthcare claims, the Data Source 110 may contain 2-4 years of historical claims data, provider master-file data and recipient eligibility data. Claims data may typically span all types of covered services including inpatient, outpatient, pharmacy, dental and long-term care data.

Detection System 120 inputs are derived from the Data Source 110. The Data Source 110 is processed through conventional training methodologies to develop the Detection System 120. The Detection System 120 may include Predictive Models 121 and/or Rules 122. The Detection System 120 is preferably a statistical model (e.g. a neural network, multivariate regression, or the like) or a procedural model (e.g. decision tree, expert system), or even a combination of both. The Predictive Models 121 may be developed using supervised or unsupervised learning techniques. The Predictive Models 121 generate predictive results (usually in the form of a score(s) with associated reasons, and a series of summary reports that will be viewed via the WebStation 150). In a preferred embodiment, the Detection System 120 is developed using the profiling techniques disclosed in co-pending application “Cascaded Profiles for Multiple Interacting Entities,” Attorney Docket No. 4790, Serial No. 09/606,575, which is incorporated by reference herein.

In one embodiment, the target of a Predictive Model 121 or Rule 122 is the entity whose fraud risk is being assessed. For example, a model may generate a fraud score for a healthcare provider, a healthcare provider-group, a licensee, a recipient or any of a number of other “entities.” When reviewing results, it is desirable for the user to know

precisely what the target is. Similarly, each Rule searches for patterns of activity that relate to a specific entity. In the WebStation150 a Target Id takes the form of an Id for the entity being assessed. The Target Id may refer to a provider number, a license number, or a recipient number, or the like. The target may even be a combination-entity such as provider-recipient pair, in which case the two Id numbers are concatenated to form the Target Id.

The results of the Detection System 120 are stored in the Results Database 140. This database may also contain pre-computed reports designed to help human reviewers determine the best actions to take in response to identified outputs (e.g. identified fraud patterns). The Database 140 serves as the communications path for delivering detection results generated by the Detection System 120 to the end users via WebStation and optional third-party reporting or visualization tools. The Database 140 provides quick access for end-users and enables execution of complex detection queries in a reasonable amount of time.

The Summarized Data Generator 130 transforms the raw data in the data source into summary data to be displayed to end-users by the WebStation 150. The summarized data is stored in a set of data tables in the Database 140. The WebStation 150 provides an interface for end-users to view the predictive results stored in the Database 140 and data from other databases such as raw or summarized data relevant to the end-user.

The WebStation 150 is a highly configurable and general-purpose interface, designed to serve as interface for many different predictive applications in many different industries. As further explained below, the user interface features of the WebStation are implemented as web pages with hyperlinks between them for navigation purposes. The WebStation 150 includes a Report Tree module 170, a Case Management module 180, a Configuration File 190, and a WebStation Builder module 195.

One embodiment of the interface deployed using the WebStation software is for a healthcare fraud and abuse detection product, hereafter sometimes called “Spyder” or “Spyder Analysis WebStation.” In the case of Spyder, the Data Source 110 contains healthcare claims, provider data, patient demographics and other supporting data. A typical Data Source might contain data for 100,000 providers, 2 million recipients, and

several hundred million claim details (where there is one detail for every medical service claimed). The scope of this data further evidences the usefulness of the WebStation for mining the data through the report tree and other features.

In this embodiment, the Dection System 120 assigns a fraud-risk score to each 5 “entity” (provider, patient, etc.), and generates explanatory reasons for each score. Here, the Summarized Data Generator 130 builds a set of data tables that summarize provider, patient and peer-group activity. (Peer groups are populations of entities that are expected to have similar activity patterns. For example, a provider peer group might contain providers who have the same medical specialty and who deliver services in the same 10 general geographic location.)

Spyder’s WebStation Database 140 contains the fraud-risk scores and associated reasons for all scored entities, summarized data, WebStation 150 user information (passwords and security information, for example), case-management information (case status and tracking information, for example), and supporting tables, such as tables to 15 associate text descriptions with medical procedure (CPT) codes, diagnosis codes, zip codes and other similar “look-up” data.

In this embodiment’s operation, end-users operate a Spyder Analysis WebStation 150 by starting the web-browser on their desktop, navigating to a Spyder URL, and logging in via Spyder’s login page. Once logged in, the user navigates from page to page 20 via hyperlinks, as with any web site. Further details of the operation of the Spyder Analysis WebStation are described in Section IV. Spyder is used as a concrete example of the one possible implementation of a WebStation 150. Other examples where the WebStation could be deployed include worker’s compensation claimant fraud detection, 25 worker’s compensation employer fraud detection, credit-card fraud detection, credit-card profitability and risk prediction, worker’s compensation care management triage, healthcare outcomes prediction or any other predictive task.

II. A Deployed Predictive Solution Using a Webstation

Referring now to Fig. 2, there are shown the two phases in which the Webstation is used, Development 200 and Execution 240. The Development Phase 200 begins with

the loading of data extracts 205 onto a hardware platform, into the Data Source 110. The Data Preprocessing 210 step transforms the raw data (claims and supporting data, for Spyder) into behavioral profiles (e.g., of providers, patients and other entities for a Spyder-type embodiment). In one preferred embodiment, the preprocessing is performed
5 using the methods described in co-pending application “Cascaded Profiles for Multiple Interacting Entities,” Attorney Docket No. 4790, Serial No. 09/606,575, which is incorporated by reference herein. These profiles are the source of model inputs 215. The Development process continues with the execution of the current model 220, generation of suspects 236 and gathering 230 of feedback 235 (assessing the quality of high-scoring
10 suspects). This feedback 235 drives the tuning of the model to ensure that it operates optimally in the customer environment.

In the Execution 240 (or Deployment) Phase, data extracts 205 (incremental feeds of new claims and supporting data, for Spyder) are fed to the Detection System 120 where behavioral profiles are computed and the predictive models and/or rules are executed to
15 generate predictive outputs (for Spyder, these are fraud risk scores for each provider and explanatory reasons for each scored provider). The scores and reasons 280 are packaged into a file for loading into the WebStation Database 140.

Data extracts 205 also feed into Summarized Data Generator 130 process that computes all of the summary information required to support WebStation operation by
20 end users. The summarized data 270 is also packaged into a file for loading into the WebStation Database 140.

Referring now to Fig. 3, on the WebStation deployment platform 300, the suspects scores and reasons 280 as well as the summarized data 270 are loaded into the System Database 330 via the Data Loader process 320. (The System Database 330 refers
25 to all data sources that are accessed by the WebStation 150. This may be one physical database or several different physical databases, each containing some of the data required by the WebStation 150.) The WebStation Server process 150 stands between end users and the System Database 330, accessing and delivering data to meet each user's

requests. Users operate the WebStation through a web browser 160 installed on their desktop.

The WebStation 150 for a particular predictive application (for example the Spyder Analysis WebStation) is a software application that is built by a partly automated and partly manual process. The automated part is done by the WebStation Builder process 195, which reads a WebStation Configuration File 190 and generates some of the WebStation software. Section VIII, "WebStation Configuration File" provides a description of this process. Manual software development then completes the specific WebStation software for each particular predictive application.

10 III. Overview of WebStation Features

The WebStation 150 includes various innovations:

1. Directed analysis of predictive results using the Reason-Driven Report Tree.
2. The WebStation frame-layout for simplified presentation of data and navigation.
3. Integrated case management functionality.

15 **Reason-Driven Report Tree**

The WebStation 150 is designed to enable users with minimal computer training to view, understand, and act on predictive results. To this end, each Predictive Model 121 has a Reason-Driven Report Tree (Report Tree). The Report Tree is a collection of reports, which are graphical and/or tabular views into the data contained in the System Database. The reports are organized into a tree, with high-level summarized data near the root, and detailed or "raw" data at the leaves. Fig. 4 shows a sample Report Tree 400 for a detection model.

20 For example, for a Spyder-type Predictive Model 121, the Report Tree 400 for a model or rule begins with the Suspect List 402—this is the "root" of the tree. Emanating from the root, for each suspect there is one branch 404 for each reason (reasons are explanations of predictive results) the suspect is included in the Suspect List. Along each reason's branch, there is a high-level summary report that displays data in tabular and/or graphical form displaying information related to that reason. These reports are preferably

pre-defined (and may be pre-computed and stored as separate tables in the Database 140). For example, one possible reason for a Spyder model is that the suspect provider delivers an unusual mix of procedures relative to his peers. The high-level report for that reason (the Procedure Mix Report) displays, for each major group of procedures, the amount of 5 activity that the suspect delivers in that group and the amount delivered by his peers.

From each report in the Report Tree, the user can follow links (e.g. through hyperlinked buttons or other user interface elements) to more detailed reports (moving from the root toward the leaves of the Report Tree). For example, in the Spyder Analysis WebStation, the user can link from the Procedure Mix Report to a Distribution Chart for 10 any of the major procedure groups. The user can also link from the Procedure Mix Report to the Patient Summary Table (a table with one row per patient that the suspect provider has treated, summarizing each different patient's past activity). From the Patient Summary Table, the user can link to the Patient Claim Detail Table (which shows the actual procedures delivered for that patient).

15 The WebStation allows users to "move around" on the Report Tree, navigating out to the leaves (most detailed reports) and then going back to the same or different high-level summary reports (closer to the root). The reports in the Report Tree are pre-defined, but the user can modify them through a variety of actions including changing the sort order of data in tables, filtering (executing a sub-query) the data in a table, changing 20 the peer group for data shown in peer-comparison tables and other similar actions.

Each Report Tree 400 has an associated Table Tree in the Database 140, listing the table included in each report node. Preferably, each table contains pre-computed results and each node in the Report Tree maps to a single table in the Database.

The Report Tree 400 allows users to easily follow a logical path of reports to 25 develop a deeper understanding of the predictive results. For example, for a Spyder-type environment, the Report Tree 400 helps users review each identified suspect provider without a detailed knowledge of the specific reports that are available. More generally, for each predictive model, this path is defined by the most important score-reasons identified for inclusion of a suspect. The reasons determine which part of the tree 400 the 30 user will explore. Along each branch the user is presented with clearly identified links to

other summaries or detail raw data. Although the Report Tree 400 defines a default sequence of reports for review, the WebStation allows users (who are familiar with the reports that are available) to view any report at any time. Section V below, "Report Tree Example" provides a detailed illustration of an actual Report Tree for one of Spyder's predictive models. In addition, Section IV, "Spyder Analysis WebStation: A Quick Introduction" also describes a report tree usage in detail.

The same Report Tree concept applies to rules when these are used. There is one Report Tree for each detection rule. The content of the reports depends upon the specific content area being analyzed (e.g. fraud and abuse scams) that the rules are designed to detect. Because rules typically target specific in the lowest level data (e.g., claims), the Report Trees for rules typically contain fewer levels than those for models. The user will typically see one summary report followed by a listing of detailed claims that caused the rule to fire.

WebStation Frame Layout

Viewing data is a central function of the WebStation. It is desirable to present data in a clear, easy to understand and consistent format. To this end, the WebStation follows a specific frame-based layout, with each frame displaying a specific type of information, and enabling users to perform a specific set of functions. ("Frame" is a web browser term with a specific technical definition in the HTML standards. To the user, a frame is a region on the screen, usually rectangular in shape with a clearly shown boundary that their browser treats as a separate region for control, viewing, printing and other functions.) The WebStation frame layout is designed to make operation of interface easy and intuitive. When presenting data within the WebStation, three frames are presented and preferably always occupy the same location on-screen. Fig. 26 illustrates the frame layout.

The Main Menu Frame 502 provides a quick way for users to move from one part of the user-interface to another. Using the Main Menu Frame 502 the user can move directly to any of the main screens of the user interface. The contents that appear in the

Main Menu Frame do not change as the user moves about the interface. The Main Menu Frame appears at the top of the browser window 500.

The Context Menu Frame 508 provides functions that make sense given what the user is currently viewing on-screen. For example, when a data table is visible in the Display Frame 504, the Context Menu Frame displays functions like sorting, filtering and changing the table layout. The Context Menu Frame appears (only when needed) at the far left hand side of the browser window.

The Display Frame 504 presents information. It is the largest frame, occupying most of the screen. What is shown in the Display Frame depends upon the function that the user is working to accomplish. Sometimes the data frame shows a table of data, sometimes it shows a graphical report, sometimes it shows a form for the user to fill-in and submit. Readability is the primary objective, so special consideration is given to color, font, spacing shading and other graphical methods that make viewing data easier.

The Display Frame shows data from a single database table (thus corresponding to a single report in the Report Tree). The way the data looks on-screen is specified by the Display Type. There are three possible display types (though others may be added as desired):

Data Grid: Shows data in tabular form.

2-By Peer Chart: Shows data as a table of bar-charts comparing individual vs. peer values. The table is defined by up to two "by" variables (such as patient age and sex). Fig. 6a illustrates a sample 2-By Peer Chart.

Distribution Plot: Shows the histogram (bar-chart) for a variable across the individual's peer-population, with a vertical line indicating where the individual falls in the distribution. Fig. 6b illustrates a sample distribution plot.

The Report Tree Navigation Frame 510 shows a recent history of the user's path within the Report Tree 400. The Report Tree Navigation Frame appears only when the user is viewing reports in the Report Tree. Icons 512 appear in this frame representing previously viewed reports in the Report Tree 400. By clicking on any of these icons, the user will go immediately to the associated report in the Report Tree. The Report Tree

Navigation Frame is a very narrow frame that appears at the bottom of the browser window.

WebStation Integration with Case-Management

The WebStation can be deployed with or without case management functionality.

5 Case management refers to the set of functions required to create, update, understand and track cases. Depending upon the specific predictive application, investigation of a case can span a range of activities from desk-audit (looking more closely at the data relevant to a specific predictive result) to field-investigation (which, for some applications, may include under-cover work, interviews, surveillance etc.).

10 A “case” refers to a suspect that has been deemed appropriate for further review or investigation. This review typically starts out as an internal analysis of the suspect’s behavior, but may at some point involve external organizations or become a legal case in a court of law. A case always has a status (open, under investigation, closed as valid, closed as fraud, etc.). A case is created when an identified suspect is first “locked” by an authorized user. Over its lifetime a case may be worked by many different users, but at any given instant, only one user can lock the case. At that instant, case-management functions (changing case status, attaching comments etc.) can only be performed by a user that has locked the case. When a case is locked it is viewable by other users but cannot be locked by another user until it is released by the first user. Each case has a unique

15 Case Id that serves as a label for the case as long as the case is stored in the system.

20

The WebStation allows case management functions to be included “seamlessly” (to the user, it looks like case-management functions are built-into the WebStation). The WebStation Main Menu Frame 502 can include case-management functions (such as “Assign Case” and “Goto Case”) and the WebStation knows when a user is in a context to perform these functions. Once the user navigates to a case, they may modify it if they have the proper authorization to do so. The WebStation guarantees that multiple users cannot modify a case at the same time.

The Case Management Module preferably provides this integration. The Case Management module tracks a variety of information about each case, including reviewer

comments, action notes and status changes. Because all case related information is time-stamped the complete history of each case can be reconstructed (who reviewed the case, when they reviewed it, what they thought about the suspect, actions they took and the status they assigned to the suspect).

5 Each case has an associated collection of fraud-risk assessments (one or more model scores and/or query results), summary reports, user comments and status updates that relate to the “entity” that has been identified as a suspect for fraud, abuse or waste. There are many different entities whose behavior may be analyzed by the Detection System 120. These include providers, recipients, licensees and provider groups. Every
10 case relates to some specific entity.

The Case Management module 180 provides functionality to:

- Manage Cases: Allow authorized users to view, open, update, track, save and close cases
- Assign Cases: Allow Management users to control who can access and review a case
- Generate Management Reports: Allow Management users to generate reports to understand current case-load, status, program effectiveness, investigator effectiveness etc.

15 In order to provide security and access control it is useful to define several
20 different user types:

- Management users are authorized to perform all functions. This includes user-management, management reporting, assigning cases to users, viewing all detection results and reports and all case management functions.
- Internal users are authorized to perform all viewing and case-management functions.
- Internal Limited users are allowed to perform all viewing and case-management functions for only those entities (providers, recipients etc.) that appear in their access list.

- External users are authorized to perform all viewing functions.
- External Limited users are authorized to perform viewing functions for only those entities (providers, recipients etc.) that appear in their access list.

5 The Case Management Module 180 further provides functional related to the management of queues. A queue is a list of suspects. A queue is defined by a set of entry criteria (conditions that have to be met in order for a suspect to be included in the queue).

A queue can serve many different purposes. For example, a queue may be defined to include suspects from a limited geographic region. Then, a Management user can
10 assign a user to the queue so that investigative work that requires travel to the same region is handled by the same investigator. Similarly, queues can be defined to distribute work across the available staff based upon medical specialty, type of fraud scheme, case status or other criteria.

In addition to distributing review work, queues can be used to implement and
15 track specific fraud and abuse strategies. For example, a queue could be defined to focus on fraud and abuse for high-volume providers. Such a queue would have two criteria, (fraud-risk score > X AND dollar-volume > Y). Using the Case Management module, Management users define new queues, and modify existing ones.

An entity is a logical unit that can be characterized by looking at a collection of
20 related data items. Examples of entities include providers, recipients, licensees, provider groups, provider-provider pairs (linked through referrals, for example) and provider-recipient pairs. The behavior of each of these entities can be captured by looking at different “slices” of the historical claims data. An entity may be the focus of a fraud detection model or may simply be used to help understand the behavior of another entity
25 (for example, looking at the behavior of the patients a provider has treated in order to better understand the behavior of the provider).

The target of a detection model or rule is the entity whose fraud risk is being assessed. For example, a model may generate a fraud score for a provider, a provider-group, a licensee, a recipient or any of a number of other “entities”. When reviewing

results, it is important to know precisely what the target is. Similarly, each detection rule searches for patterns of activity that relate to a specific entity.

Case-management functions are described in Section VII, "Case Management Module Software", along with illustrative aspects of the case management user interface.

5 Additional case-management concepts (such as the Web Casebook) are described in Section VI, "WebStation Functional Description." The seamless integration of case management functions is shown in the screenshots described in Section IV, "Spyder Analysis WebStation: A Quick Introduction".

IV. Spyder Analysis Workstation: A Quick Introduction

10 This section provides a quick introduction to the Spyder Analysis WebStation embodiment of the present invention. This embodiment focuses on using the features of a WebStation in a predictive solution for analyzing healthcare fraud and abuse; other applications of the WebStation are certainly possible. The goal is to illustrate the basic functionality and show how a WebStation embodiment can be used to review suspects
15 identified by the Detection System 120.

A Browser-Based Approach to Viewing and Analyzing Detection Results

The WebStation is a browser-based application, as such it requires no desk-top software installation beyond a conventional browser, such as Netscape Navigator (version 4.x or higher) or Internet Explorer (version 4.x or higher). Users who access the Internet
20 via one of these browsers should find operating a WebStation 150 fairly intuitive. Like web sites on the Internet, the WebStation 150 uses links on one page to get to other pages. In the WebStation 150, these pages contain information about suspects identified by the Detection System 120, which the Spyder embodiment are models and rules pertaining to healthcare fraud and abuse.

25 The WebStation 150 is designed to help users take the first step in analyzing detection results. Users may or may not be able to pinpoint the specific claims that are inappropriate, abusive or fraudulent, but they should be able to determine whether or not more detailed investigative work is required, or if there is a valid explanation for the

suspect's activity. To accomplish this, the WebStation 150 presents suspect lists, and enables users to view a wide array of summary reports describing the suspect. These reports provide insight into the suspect's activity at many different levels of granularity, from high-level summary to claim line-item detail. The claims data, suspect-lists and 5 summary reports that the WebStation presents on-screen are housed in a database 140.

A Sample WebStation Session

User Log-on

Referring now to Figs. 7-16, there is shown a sample user session with an embodiment of the WebStation. A WebStation session begins when the user logs on. 10 The logon page (Fig. 7) is the first page presented and requires users to enter their user-ID and password. Each WebStation user has an assigned access-level that determines which functions they can and cannot perform. For example, only a Management-Level user can assign cases to other users, add new users to the system or change the access-level of an existing user.

Primary Detection Page

The first page (Fig. 8) that WebStation users see is the primary detection page 800 (sometimes referred to as the "home" page) which lists the detection models 810 and rules 820 that have produced a list of fraud and abuse suspects. Because Spyder suspects can be providers, patients or other "entities," the predictive models 815 and rules 830 are 20 organized by the type of "entity" they detect.

A color-coded bar at the far left of the screen ("Provider Detection" 840) indicates the type of suspect (here "providers") that is being assessed by the detection models and rules. When the user selects the "Description" button 850, a text description of the model or rule is shown. (Rules for patients and other entities are not shown in this example.)

25 The WebStation presents information in "frames." When a user looks at a screen, they will typically be looking at several different frames as described above with respect to Fig. 5. The screen shown in Fig. 8 includes two frames, one (the Display Frame 504) shows the lists of available models and rules, the other (the Main Menu Frame 502)

appears as a set of “tabs” 860 at the top of the screen. The WebStation shows these same tabs on all subsequent screens. These tabs operate like a “main menu” allowing users to select the major functions that the WebStation provides, no matter where they currently are within the WebStation.

5 o The WebStation’s “main menu” functions include:

Menu Tab	Function Description
Home 870	Enables users to quickly return to the “home” page that lists detection models and rules.
Assign-Case 872	Allows Management users to open a case for a given suspect and assign it to a user or group of users.
Search 874	Allows users to search for individuals by entering identity information (ID number, first name, last name, etc.) for specific “entities” (providers, patients etc.). The result is a list of entities that “match” the search criteria. The user can then view summary data for that entity.
Maintain Users 876	Allows Management users to add new users and change properties for existing users (such as access level).
Reports 878	Allows users to view specific reports by selecting the report from a list of reports.

Go To Case 880	Allows users to view a case for the suspect currently shown on-screen. The suspect currently being viewed must have a case opened and the user must have permission to view the specific case.
Log-off 882	Logs the current user off and allows a new user to log on.
Options 884	Allows users to perform secondary functions such as changing their password, defining the colors and fonts to be used in the interface, etc.
Help 886	Opens a separate browser window that provides a help facility.
Resources 888	Displays internet links to helpful healthcare fraud and abuse sites.

The Suspect List is the “Root” of Spyder’s Report Tree

There is one suspect list for each predictive model 815 or rule 830. This suspect list forms the “root” of a reason-based report tree, such as illustrated in Fig. 4. As the name suggests, this is a collection of reports, organized into a tree. The suspect list is the root, then, for each category of explanatory reason, there is a different branch.

The first report along each branch is a high-level summary that presents an overview of activity related to that branch’s reason category. As the user moves down the branch, they can select specific reports to view. The reports near the root are high-level summaries, while the reports near the end of the branches (the “leaves”) are very detailed, usually showing detailed claims.

Users view a specific suspect list by selecting the name of the model 815 or rule 830 from the WebStation “home” page 800.

Fig. 9 illustrates a sample suspect list 900 which would be the root of report tree. The suspect list 900 shown in Fig. 9 includes:

- Suspect-Id 902 (for this model, suspects are providers)
- Score 904 (the higher the number, the greater the fraud risk)
- Score-Date 906 (indicating when the model was executed)
- Rank 908 (1 is the highest-scoring provider)
- 5 • The suspect's total dollars paid 910
- The suspect's number of claim details 912
- The suspect's number of patients 914
- A set of explanatory reasons 916 (which indicate why the model produced
the score that it did for each provider). The reasons are ordered based
10 upon relative importance to the score

Notice that two additional frames appear. The Context Menu frame 508 on the far left, and the Report Tree Navigation Frame 510 at the bottom of the screen. The Context Menu frame 508 enables users to perform a variety of useful functions on the data that appears in the Suspect List. These same functions are available any time the
15 WebStation displays data in tabular form. A closer look at these functions is made later; this section describes how a user can select a suspect to look at, and then drill-down from high-level summary to claims detail for that suspect.

As previously mentioned, the Report Tree Navigator frame 510 at the bottom of the screen displays small icons 512 that represent the report pages that the user has
20 visited. Notice that only one small icon appears in the sample above. This icon represents the suspect-list 900. As the user drills down into additional reports (following a branch of the report tree), additional icons will appear in this frame. If the user wants to “back up” and revisit a previous report, they can do so by clicking on the icon representing that report.

25 **Viewing Spyder's Standard Summary Reports**

Focusing again on the screen showing the suspect-list 900, if the user clicks on a specific Suspect-ID 902 (which is a Provider-ID in this example), the WebStation 150 takes the user to a standard summary for that provider. Notice that the data fields that

serve as hyperlinks follow the Internet convention and are shown in color-coded, underlined text.

The Provider-ID appears on many different pages within the WebStation, but selecting the Provider-ID always takes the user to the standard summary for that Provider.

5 Fig. 10 illustrates the format of one version of a Provider Summary page 1000. The Provider Summary 1000 includes separate blocks of information about:

- Identifying information 1002 (provider numbers, license number, name, address etc.)
- Fraud-risk results 1004 for all of the detection models and rules that apply to the provider.
- General statistics 1006 to indicate the size and type of the provider's practice
- The age/gender mix 1012 of the provider's client-base
- The mix of procedures 1014, 1016 the provider delivers

10 15 User can view similar standard summaries for other entities (such as individual patients). The method for accessing these summaries, as with the Provider Summary, is by clicking on the ID for the specific entity as it appears in any data table within the interface. After viewing the Provider Summary, the user returns to the Suspect List 900 by using the "Back" button on the web browser.

20 25 If, from the Suspect List 900, the user selects a specific reason 916 for a given suspect, the WebStation 150 takes the user to the high-level summary report that captures the "essence" of that reason. Reasons serve as explanations for model fraud-risk scores. Each suspect has several reasons, which are listed in order of importance. Reasons are different for each model. The example used here is a model that detects fraud and abuse for Dental Providers.

A typical detection model will have 10-20 reason categories. Each category has links to a high-level summary report from the Suspect List 900 for that model. Fig. 11 is an example of the type of high-level summary that the WebStation displays when the user selects a reason from the Suspect List.

This report is called an "Abnormal Clinical Procedure Mix" report. In this graphical report, the WebStation displays a summary (upper charts) of the provider's activity in each of 11 different clinical procedure groups. The mix across groups is shown both in terms of dollars paid and number of services. For each procedure group, 5 the individual provider is shown as a dot 1110 (here a half-circle because of its placement on the X-axis and on the 100% line), while the provider's peers are shown as 3 diamonds 1125 (connected with a thick line). These three diamonds (which sometimes are so close together that they appear as a single diamond) represent the median (middle diamond), 25th percentile (lower diamond) and 75th percentile (upper diamond).

10 Where the provider dot 1110 appears with respect to the 3 peer-group diamonds 1125 tells the user where the individual falls with respect to their peer group. Thus, in a glance, the user can tell if the provider is doing more or less of a particular type of service than his peers, and can immediately see if the difference is large or small, relates to a single category of services or to several categories, etc.

15 The two bar charts at the bottom of the figure show the individual provider's top five CPT codes both in terms of dollars paid and number of services. By clicking on the "Show Provider/Patient data" button at the bottom of the screen, the user can proceed to the next level of detail, as illustrated in Fig. 12.

In this report 1200, there is one line 1202 per patient (it lists the patients for which 20 the suspect provider has delivered at least one service). Each line indicates the patient's ID 1204, Age 1206 and Gender 1208, as well as specific aggregate measures of the services received by this patient (average dollars per service 1212 and total dollars 1214, and number of services 1216) both from the suspect provider and all other providers (for which there is claims data available). From this report, if the user selects (e.g. clicks on) 25 the number of services 1210 for a patient or the patient ID, the WebStation takes the user to a list of that patient's claim details. Fig. 13 illustrates this report 1300.

While not shown here, the main menu tabs 870-888 as shown in Fig. 8 are preferably always shown on-screen, and the navigation icons 740 are always shown in the Report Tree Navigation frame 510 along the bottom of the screen. The list of functions

(filter, sort, layout, etc.) are shown in the Context Menu frame 508 at the far left any time data is displayed in tabular form.

The claim detail page is a good place to show how some of these functions work. Referring to Fig. 14, to select a function, the user clicks on the desired function (e.g.

5 Filter 1401) in the Context Menu frame 508 at the far left of the screen. Filtering allows users to execute queries that qualify which rows of the current data grid should be shown on-screen. For example, as shown in Fig. 14, if the user is only interested in services where the amount billed is greater than \$30.00, he selects “>” 1404 in the \$ Billed column 1402 and enters the amount “30.00”. Each column permits multiple criteria, and

10 the user can enter criteria for as many columns as he likes. To execute the filter, the user selects the “Apply” button.

Referring to Fig. 15, when the user selects the “Sort” function 1501, the current sort order is displayed and the user is given the opportunity to specify a new ordering. Numbers 1504 indicate the sort hierarchy, and arrows 1506 indicate increasing (up) or decreasing (down). In the example of Fig. 15, the sort order is first by decreasing \$ Billed and then by date-of-Service (most recent first). Fields not included in the sort definition remain unnumbered (indicated by a small box 1502 at the top of the column). Clicking on this box 1502 includes the column in the sort criteria, clicking on a number removes the column from the sort criteria. Clicking on the up or down arrow toggles the direction of the sort. When the user has defined the desired sort order, they can execute the sort by clicking on the “Apply” button.

When the user selects the “Layout” function 1601, the current layout is displayed (Fig. 16) and the user is given the opportunity to change which fields are displayed as well as the order in which they appear on screen. Right/Left arrows 1602 move fields between the “Shown Columns” 1606 and “Hidden Columns” 1608 lists. Up/Down arrows 1604 move the selected field up or down in the “order of appearance” indicated in the “Shown Column” list. The “Apply” button executes the layout changes.

The Graph function 1607 allows users to pick which columns of data they would like to use for a plot, and the type of plot they want to view (scatter, pie, line, bar, etc.).

The WebStation computes and graphs the selected plot using the data from the selected columns. Sample plots are shown in various ones of Figs. 17-27.

V. Report Tree Example

This section describes in further detail a sample report tree, as may be generated by the WebStation for analyzing in more detail the relationships of various entities to each other. The illustrated reports are schematic in format and not representative of how they would actually appear on a screen display, though some of portions of the figures illustrate features of such screen displays. In addition, the linkages between reports are explicitly illustrated in many cases to show how the user navigates between reports. While this example is specific to a healthcare fraud and abuse implementation of the WebStation and the report tree concept, the report tree may be utilized in many other types of applications, including other types of fraud and abuse systems (e.g., consumer, merchant, Internet, credit card, debit card, etc.), sales and marketing analysis system, or any other application where multiple entities interact with each other in complex data relationships.

Procedure and/or Diagnosis Groups (TOS and/or POS) (Fig. 17): This figure shows a group of reports for a reason category that pertains to a break-down of the provider's activity in a particular grouping scheme. The four grouping schemes for the MD/Physician model are - Procedure Code Groups, Diagnosis Code groups, Type of Service (TOS) codes and Place of Service (POS) codes. For a reason corresponding to any of these grouping schemes, the reports in Fig. 17 will show up, with the corresponding groups.

First, the report header 1702 shows the grouping scheme, the provider number 1704 being reviewed and the peer group 1706 that is being used for the provider. The peer group can be changed using a drop-down box 1708 - this will change the set of providers against which this particular provider's statistics are being compared. Then, the percent of the provider's activity (by dollars paid and by number of services) in each group is shown. (These are the top two images titled "DOLLARS PAID" 1710 and "NUMBER OF SERVICES" 1712). For each group, in addition to showing the level of

the provider's activity in that group (depicted by the dot, e.g. dot 1714), the 25th, 50th and 75th percentile values for percent activity in that group across all providers in the selected peer group is also shown. This is depicted by the 3 horizontal lines across the vertical bar for each group (e.g. bars 1716). When the user clicks on the dot showing the provider's level in a particular group, this takes the user to a distribution report that is illustrated in the Provider (Yellow Dot) Chart in Fig. 19. The two bottom charts (1718, and 1720) of the Procedure and/or Diagnosis Group report show the top 5 codes in which the provider's activity is concentrated, one by dollars paid and the other by number of services.

The bottom of this report contains 3 buttons. The "View Client Summary" button 1722 takes the user to a report showing a summary of the providers' activity (relevant to that grouping scheme) with each of his/her clients (one line per client). For example, for procedure code groups, the number of details in each procedure code group for each client will be shown. Clicking on any one line, will takes the user to the claims corresponding to that client. The "Flip Client Summaries" button 1724 takes the user to the report shown in Fig. 18. The "View Procedures" button takes the user to the Provider (Yellow Dot) Chart report, shown in Fig. 19.

Flip Client Summaries (Fig 18): In drill-down mode, the user comes to this report by clicking on the "Flip Client Summaries" button 1724 at the bottom of Procedure and/or Diagnosis report. This report allows the user to "flip" through the summary of a provider's activity with any given client and compare it to the activity of all other providers on that client. Once again, the user can select the peer group in the drop-down box 1804 in Report header 1802. The "Pick Flip Set" drop-down 1806 allows the user to select the grouping scheme (e.g. Procedure, Diagnosis, TOS, or POS) and number of services or dollars paid for which the user is interested in seeing the activity break-down. Corresponding to the scheme selected, the bar graph 1808 shows, the level of the provider's activity by the different groups compared to all other providers' activity, for the given client. The "Basics" box 1810 shows the basic client summary report (one line per client), off of which a particular client of interest may be selected. The box 1812 next

to that shows the statistics for the selected client's activity, across the various groups, for selected providers vs other providers.

Provider (Yellow Dot) Chart (Fig. 19): In drill-down mode, the user comes to this report either by clicking on a yellow provider dot 1714 on the top report on the Procedure and/or Diagnosis report, or by clicking on the View Procedures button 1726 at the bottom of that report. The top chart 1906 on this report illustrates the distribution of the selected statistic (e.g., % activity in group N) across all providers in the selected peer group (once again, selectable through the drop-down 1904 in the report header 1902), and shows where the value corresponding to the current provider of interest lies on that distribution (the vertical line 1914 in the chart). The buttons 1916, 1908 below the chart allow selection of appropriate subsets ("Select Subset") and viewing the distribution within the selected subset ("View Subset"). The third button 1910 "Show Table" takes the user to table 1912. This table has one line per procedure code (or diagnosis code, etc) that the provider has activity in. For each procedure code, the activity by number of services or dollars paid and the corresponding percents is shown for the selected provider.

Client Age Breakdown (Fig. 20): This figure shows reports for the reasons corresponding to breakdown of a provider's clients by age groups. The top three reports 2004, 2006, 2010 respectively show the breakdown of numbers of clients, number of services and the dollars paid for the selected provider by the different client age groups. These charts are similar to the ones described for the Procedure and/or Diagnosis report, where the yellow dot represents level of selected provider's activity and the horizontal lines show the 25th, 50th, 75th percentiles of that activity for the peer group. Again, clicking on a yellow dot in a particular age group takes the user to a distribution chart 2016 similar to the one described for Fig. 19, the Provider (Yellow Dot) Chart report, except that the chart 2016 will show the distribution of services only for the selected age group. The View Client Summary button 2012 once again takes the user to a one-line-per-client report, showing the basic statistics for that client for the selected provider. Clicking on any line, will take the user to the corresponding claims.

From the distribution chart, the View Client Summary 2012 button will take the user to the same report as the corresponding button 2012 on the top chart, except now in a

specific age group. The View Breakdown button 2020 will take the user to a Procedure and/or Diagnosis report like Fig. 17, except that the activity shown will correspond to the particular age group that was selected (through the yellow dot). The third box 2022 enables selection of breakdown by different codes/groups.

5 Monthly Activity (Fig. 21): This report is for reasons corresponding to monthly activity of provider. The breakdown of activity across months is shown for number of clients (2106), number of services (2108) and dollars paid (2110). Again, each yellow dot (e.g., dot 2112) represents level of selected provider's activity and the horizontal lines (e.g. 2113) show the 25th, 50th, 75th percentiles of that activity for the peer group. Like
10 before, clicking on a yellow dot will take the user to a distribution Provider (Yellow Dot) chart as in Fig. 19. The Select Month button 2114 at the bottom of each chart enables selection of a particular month for which the provider's claims, sorted by client are shown. The Show Breakdown button 2116 enables viewing breakdown by different codes/groups, showing Procedure and/or Diagnosis reports similar to ones illustrated in
15 Fig. 17, but for a particular month of activity.

Client Consecutive Visits (Fig. 22): This report is for reasons corresponding to the periodic nature of clients' visits for the selected provider. Top left chart 2206 shows the percent of clients who have consecutive monthly visits for 1,2, . . . 12 months, while the bottom left chart 2210 shows the percent of clients who have more than two visits per month consecutively for 1,2, . . . 12 months. Again, each yellow dot (e.g. dot 2213) represents level of selected provider's activity and the horizontal lines show the 25th, 50th, 75th percentiles of that activity for the peer group. Like before, clicking on a yellow dot will take the user to a distribution chart 2208 similar to the one described for the Provider Chart in Fig. 19. Once again, the user can go down to the corresponding
20 Client Summary report 2214 from these charts and then to the corresponding claims. The buttons 2212 at the bottom of the charts also allow the user to select clients for a particular number of months and then look at their summary and claims reports.
25

Group Participation (Fig. 23): This report is for a reason corresponding to the participation of the provider in a group. Here, the provider's group 2302 is identified and

likelihood of peers being in a group 2304 is shown. This report can link to clients' statistics and claims for that particular group 2306.

Dollars Per Claim (Fig. 24): This report corresponds to the provider's distribution of dollars paid per claim. The top chart 2402 is a distribution chart like Provider Chart on in Fig. 19, showing the distribution of average dollars per claim 2406 within the peer group and where the value for the selected provider lies in this distribution (vertical line 2404). Again, the Select Recip Subset 2408 allows selection of a subset of clients by age/gender etc., by which the distribution (chart 2416) may be viewed and the Breakdown button 2410 allows breakdown by different codes/groups (similar reports to the Procedure and/or Diagnosis report, but limited to the selected subset). The distribution chart 2412 shows the distribution of the dollars/claim for this particular provider. Again, the Show Client Summary button 2414 takes the user to a one-line-per-client report sorted (by default) by \$/claim. This then leads to the claims corresponding to any selected client.

Per Day Activity (Fig. 25): This report shows distributions of the per-day activity of a provider. The charts 2502 show the different distributions as labeled. The "View Day Summary" 2504 enables viewing the clients/claims 2514 on a particular day. The View Client Summary button 2506 causes the display of a table 2512 with a line per client sorted by average dollars per client-day. The GAB 2508 and "Breakdown" drop-down 2510 enables viewing of the distribution charts by different codes/groups. The client summary leads to the corresponding claim details 2516.

Per Client Volume (Fig. 26): This report shows per-client statistics for a provider. Distributions 2602 of the different quantities for the selected provider are shown (e.g. dollars/client 2602a, number of clients 2602b, number of services per client 2602c, provider's dollars paid per client 2602d, and provider's number of services per client 2602e). Again, the View Client Summary button 2602 takes the user to a report 2606 with a line per client sorted by dollar such that clients with largest dollar volumes are at the top. Clicking on a client in that report 2606 then leads to a report 2608 of the claims for that client. One can also change the breakdown (procedure code, diagnosis codes, place of service codes, type of service codes) and the data subset (age, gender, age/gender).

Multiple Physicians Per Day (Fig. 27): This report shows the activity of a provider's clients seeing multiple physicians in the same day. Average 2702 and max number of providers 2704 seen by a provider's clients on the same day that they see this particular provider are shown. Again, the "View Client Summary" 2706 takes the user to a report 2710 with a line per client with a column 2714 showing the total number of providers seen by selected client. Clicking on that number takes the user to the claims 2712 for that client for all providers.

In summary then, it can be seen from these examples that the report tree has a general structure along the following lines:

- 10 (1) Suspect List
- (2) Breakdown of Provider's activity by procedure code groups
- (3) Distribution chart showing peer activity vs provider
- (3) Subsetting activity breakdown by age/gender
- (4) Subsetting distribution by age/gender
- 15 (5) Table showing activity by breakdown (e.g. Procedure Code, Diagnosis Code, Type of Service, Place of Service)
- (3) Client Summary report (one line per client)
- (4) Flip Client Summaries (graph client summaries and compare to other providers)
- 20 (5) Claims Report

From any given report, one can branch out into different reports, hence there are multiple reports at the same level (indicated by numbers in parentheses). Also, different report branches will have different numbers of reports at the different levels. These various reports may be described as follows:

- 25 1. Suspect list Report : This is the root of each hierarchy. It contains a list of all the providers scored by the model, rank-ordered by the model score, and showing the top reasons for each provider. Clicking on any of the reasons takes the user down the branch of reports corresponding to that reason.

5

10

15

20

2. Provider-Based Report(s): This is the first level in any of the reason branches. A provider-based report uses only the selected provider as the key. This report will show the statistics relevant to the particular reason for the selected provider. Usually the distribution of those statistics for the peer group will also be shown in some form for context. Note that there could be multiple provider-based reports on a given reason branch. For example, the first report may show a coarser level of statistics and the ones leading from it will show finer levels of statistics.
3. Provider/Subset Based Report(s): From a provider-based report, one can drill down to a report that breaks down that provider's activity by a particular subset of clients/claims. Examples of subsets include age/gender, particular codes/groups such as procedure codes, diagnosis groups, etc.
4. Provider/Client Based Report(s): This is the next level in the hierarchy of reports. A provider/client based report breaks down a provider's activity across all of the clients that the provider has seen. The coarsest level of a provider/client report is one where there is one line for each client and basic activity statistics between the selected provider and that client, pertaining to the reason of interest, are shown. This report can then lead to reports showing more data on the particular provider-client pair's activity.
5. Claim Detail (Transaction) Reports: This is the last level in the hierarchy. This will show the subset of claims corresponding to the data that was viewed in the higher levels of the report branch.

Note that this description of report applies to a provider-based model and data. A similar hierarchy may be constructed for models in other applications as well, however the specific entities involved will change. The general structure of the "Suspect List" being the root of the tree and the transactions being the last nodes in the tree holds for any application.

VI. WebStation Functional Description

This section describes the preferred general functions of the WebStation 150. The functionality related to case management features is mentioned here, and described in further detail in the next section.

Viewing Report Tree Supporting Data

5 The WebStation 150 provides a robust set of functions for viewing results of the Detection System 120 as report trees and the underlying data in the Database 140.

- 10 1. View primary detection page: Display list of Predictive Models 121 and Rules 122 with high-level summaries of their results (number of suspects, estimated dollars at risk etc.). Fig. 8 is illustrative. This summary page 806 preferably includes:
 - a) The date when the model or rule was most recently executed.
 - b) For rules: the target (provider, licensee, recipient, etc.), number of claims, providers, groups, license numbers, recipients and dollars billed. (Note that for some rules, certain summary data may not be applicable).
 - c) For models: the target, score range, number of providers, groups, license numbers or recipients in the top percentile of scores, top decile of scores(to give an idea of the scope of the model).
- 15 2. View a text description of each Predictive Model 121 and Rule 122: From the primary detection screen 800 the user should be able to view a detailed text description of each detection model and rule. This text preferably indicates:
 - a) What the model or rule does (on-line description)
 - b) The target (providers, licensee or recipients)
 - c) The detailed scope (provider sub-specialty, specific fraud scams etc.)
 - d) The data that is required in order to run the model or rule
 - e) When it was first developed
 - f) The history of modifications made.

3. View suspect lists 900 and reasons generated by Predictive Models and Rules: For models, when a model is selected, a list of suspects should be displayed using the frame layout.

4. Drill-down to view pre-defined summary and detailed reports that relate to top reasons (Reason-Driven Analysis).

5. Select from entire list of predefined summary and detailed reports.

6. Select a suspect and drill-down to the relevant summarized information, ultimately to the relevant claims detail (and other raw) data.

10 7. Search for all relevant detection results for a given provider, group, license number or recipient.

8. Search for cases by case-id, provider, group, license number or recipient.

15 9. Specify and apply arbitrary sort criteria to on-screen data as it appears in a Data Frame. Fig. 15 is illustrative.

10. Specify and apply constrained filter criteria to on-screen data as it appears in a Data Frame. Fig. 14 is illustrative.

11. View Provider Summary information via Provider Id link. Fig. 10 is illustrative.

20 12. View Recipient Summary information via Recipient Id link.

13. View Provider Detailed claims data from the Provider Summary screen.

14. View Recipient Detailed claims data from the Recipient Summary screen. Fig. 13 is illustrative.

25 15. Select a suspect and drill-down to the relevant summarized information, ultimately to the relevant claims detail (and other raw) data.

Much of the typical user's work boils down to viewing data in an effort to validate suspects and determine the nature of the questionable activity so they can take the appropriate actions. Report Trees help users navigate through the multitude of possible reports so they can quickly accomplish their objectives. User is able to start with a

suspect list and drill-down to view pre-defined summary and detailed reports that relate to top reasons (Reason-Driven Analysis) or rules (Rule-Driven Analysis). Figs. 17-27 describe examples of each of the reports and identify fields within each (non-leaf) report that link to additional reports. The user is able to initiate this drill-down through the 5 Report Tree 400 either from a suspect list 700, or from the target summary report in the following manner.

- a) Select from entire list of predefined summary and detailed reports

In addition to Reason-Driven Analysis, the user is able to 10 select from a list of available reports. This list is presented on-screen in a hierarchical (tree) structure.

- b) Search for all relevant detection results for a given provider, group, license number or recipient

The user can view all suspect list entries for any valid 15 provider, group, license or recipient by specifying the appropriate Target Id. The results identify which model or rule identified the search target as a suspect and includes all of the suspect list fields (score, score date, reasons etc.)

- c) Search for cases by case-id, provider, group, license number or recipient

The Management user can search, by case, provider, group, license number or recipient ID, for all cases that exist for 20 that entity. The resulting list includes entity ID (provider, recipient etc.), case ID, case status, case create date, last activity date.

- d) View Provider Summary information via Provider Id link

From any Data Frame including the provider ID, the user 25 can link to the standard summary data for providers. This summary includes provider master-file information (name, address, stated specialty etc.) as well as derived summary

30

data (claims volume, dollars billed, etc.). The contents of the Provider Standard Summary Report are described above with respect to Fig. 10.

5 e) View Recipient Summary information via Recipient Id link

From any Data Frame including the recipient ID, the user can link to the standard summary data for recipients. This summary includes recipient eligibility file information (name, address, eligibility start/end dates etc.) as well as derived summary data (claims volume, dollars billed, etc.). (Similarly, the user should be able to link to the target summary for other targets such as licensee, etc.)

10 f) View Provider Detailed claims data from the Provider Summary screen

15 From the Provider Standard Summary the user can link to the detailed claims submitted by the provider. The user can filter these claims by start and end dates, types of claims, etc. before retrieval (filters are available on all fields after retrieval via the Function Frame).

20 g) View Recipient Detailed claims data from the Recipient Summary screen

From the Recipient Standard Summary the user can link to the detailed claims submitted for services rendered to the recipient. Fig. 13 is illustrative.

25 h) Specify and apply arbitrary sort criteria to on-screen data as it appear in a Data Frame

Allow users to enter sort criteria by column (for example, by entering a signed number in a form, 1 = primary sort key, ascending, -2 = secondary sort key descending etc.).

Allow the user to apply the sort. Fig. 15 is illustrative.

5 i) Specify and apply constrained filter criteria to on-screen data as it appears in a Data Frame

10 Allow users to specify simple criteria (for numeric fields, constrained to >, <, =, <=, >=, with AND, OR, for example) for each column in a Data Frame. Allow users to apply the specified criteria to filter the rows of data within the Data Frame. Fig. 14 is illustrative.

15 16. View current case-list (the case-list is specific to each user and includes both cases that have been previously worked and cases newly assigned to the user). Each internal or Management user will have a list of cases that they have worked or that have been newly assigned to them (by a Management user). Users should be able to see a list of these cases with summary information relating to each case:

20 a) Case Id
b) Target Id
c) Case create date
d) Case create User Id
e) Last action date
f) Last action User Id
g) Current Case Status

25 17. Frame Layout and Content
Navigation Frame content:

Standard Elements, “View Model <model id>“ highlighted, Detection subtree detail.

Function Frame content:

View, Sort, Filter, Title, Graph elements

Data Frame content:

For Models each data row contains:

a) Target Id, linked to appropriate summary report

- b) Score Date
- c) Score
- d) Top <n> reasons, each reason linked to the Reason-Based Report Tree
- e) Link to score history report

5 For Rules each data row contains:

- a) Target Id, linked to appropriate summary report
- b) Run Date
- c) Number of claims
- d) Number of recipients
- e) Total dollars at risk
- f) Link to Rule-Based Report Tree (may be a single claim-level report)

10

Produce Hardcopy and Electronic Reports

- 1. Print any report (as it appears on-screen) relevant to a specific case
- 2. Print/Save reports as they appear on-screen
- 15 3. Allow users to print and/or save any report that is viewable on-screen.
- 4. Provide management reports to summarize the performance of specific detection models, rules or human reviewers
- 5. Allow Management users to view a summary of the case status (open, closed valid, closed fraud etc.) by detection method (Model Id or Rule Id), and by User Id (opening User Id, closing User Id).

20

Provide Security and Access Controls

- 1. Require each user to login to start a session
 - a. At the start of a session, require each to enter their Id and Password.
- 25 2. Enforce security by user access level
 - a. Assign each user a User Id, Password and User-Type (security level). Enforce security to prevent viewing of results (at the

provider and recipient level) and modification of case data by unauthorized users.

3. Allow Management users to add, delete and modify system User Ids, Passwords and User-Types. Allow Management users to enter new users and specify their User Id, User Type and initial Password. Allow Management users to delete and modify any User Id, change any Password, change any User-Type.
4. Allow Management users to assign view and modify privileges to system users at the provider-id and recipient-id level (for "Limited" user types). Provide a screen with a form to allow Management users to associate provider or recipient Ids to User Ids for "Limited" users. Allow Management users to delete provider and recipient Ids from a user's access list.
5. Allow each user to change their own password
6. Provide security to prevent viewing of results (at the provider and recipient level) and enable only authorized users to modify case status

Manage Cases

1. Open a case by "locking" a suspect from a suspect list
2. Assign a case by selecting a suspect from a suspect list

An authorized user (User-Type = Management) may assign a case to a system user. This procedure locks the case (to ensure that other Management users cannot concurrently assign the case).

3. Lock a case to start a case session, release a case to end a case session
Ensure that no other user can lock the case while it is locked. Allow only the user who has locked the case to modify case data. Allow a user to release a case. Automatically release a case when a user exits the application (leaves the site).

4. Enter comments, set case status, and take actions with respect to the case. Allow a user to enter comments and set case status only when they have the case locked.
5. Allow Management users to release any locked case.

5 Because internal users may unknowingly keep a case locked (and go on vacation etc.) Management users need to be able to release any case (as identified by the Case Id).

6. Keep track of prior actions and comments saved for each case. Allow a summary of prior actions to be viewed.

10 Make an entry into the case-action log every time a user modifies case data. Note the User ID, Date, and Start/End time for the user's case session. Allow authorized users (Internal and Management) to view the Prior Actions log.

Additional WebStation Features

The WebStation also offers users additional functions to manage users, integrate with a Case Management Module and customize the way the WebStation displays information to meet each user's needs. The following shows the supported functions, which are easily implemented by those of skill in the art:

- Password
- Sizing
- Colors
- Fonts
- Miscellaneous
- System
- Reminder Message

25 **Optional Features**

Web CaseBook

As a case is worked users perform different analyses, some using the Spyder Analysis WebStation, others taking advantage of external analysis tools and investigative

processes. One mechanism for documenting a case is the Web CaseBook. As the case evolves, different users can add reports generated within the Spyder Analysis WebStation to the Web CaseBook. Preferably, the Web CaseBook has a hierarchical (outline) structure, but a single-level list of reports, comments and analyses may be provided.

5 Each item in the list is stored as HTML and is linked to a title-entry in the main page of the CaseBook. At any point during the life of a case, the Web CaseBook can be viewed using a web browser. Preferably a CaseBook editor is used to help users develop cleaner looking Web CaseBooks.

Because predictive models and rules are executed against data from a changing
10 data warehouse, a thorough record of what the data looked like at the time of investigation is desirable. The CaseBook meets this need. Generally, the Web Casebook is a web-site for a particular case. Some users create and modify the Web casebook, while others can only view its content (with components of the Web Casebook viewable depending upon the viewing user's authorization level). Additional tools (outside of the
15 WebStation) may also provide the ability edit, update and view Web Casebooks.

Enhanced Graphics

The Display Types available for showing data on-screen may be enhanced. New Display Types can be provided for more sophisticated graphics to help users interpret the summary and detailed data needed to understand detection results. Specific new Display
20 Types may include: Pie Charts, Temporal Trend Charts and Scatter Plots.

Graphing Function

In addition to fixed graphical Display Types, an embodiment may include a user-driven capability to select columns from Data Grid displays and generate graphical plots on-the-fly.

25 **File Export**

To facilitate export of summary and detailed report data to 3rd-party packages, an embodiment may include a capability to export specified data in an easily transported

format(s). The export function supports standard file formats, such as Microsoft Excel, tab-delimited, and so forth.

VII. Case Management Module Software

This section describes in further detail the general and specific functionality of the
5 Case Management Module 180.

Functional Overview

The following is a list of high-level functions for the Case Management
Module 180.

Manage Cases

10 The Case Management Module 180 provides a number of interconnected screens
that are used to review and analyze cases. Fig. 28 shows how these screens relate to each
other:

Each user has a personal Case-List 2802. An example case list is shown in
Fig. 29. Management users build and modify this list by assigning cases to users. Users
15 will begin their session by either going directly to their case-list, or by browsing reports
and detailed data using the Analysis WebStation.

When users elect to view their Case-List 2900, they will see all of their cases,
including newly assigned cases. The number of new messages 2902 related to each of
their cases will be displayed on their main case-list page. By following the Messages
20 link, the users can read all new messages for the case. A sample of the new messages
screen is shown in Fig. 30. The user can also send messages to other users who have
authority to access the case. All messages are stored and tracked for inclusion in the case
message log.

From the Case-List page 2900, users can elect to go to a case by following its
25 Case Id link 2904. This operation takes the user to the Case Summary page (Fig. 31) for
that case and locks the case so that no other user can simultaneously modify the case.
The Case Summary page shows an overview of all of the critical aspects of the case. The
information blocks Identity 3102, Update Information 3104, Detection Sources 3106,

Attached Claims 3108, Linked Cases 3110, and so forth, group case information into higher-level categories for ease of viewing.

When the user has a case locked they can perform functions such as changing the case status, adding comments to the case notes, and other functions that modify case data.

5 When the user is finished, they exit the case, and the case is released so that other authorized users can work the case.

When browsing reports via the WebStation 150, users may also elect to go to a case. This operation is only available when 1) the user is on a page in the WebStation where a single Target Id. is clearly defined, and 2) the user has authorization to work the 10 case for that Target Id (this implies the case exists). Management users can create a case in this context. Clicking on the Go To Case tab 880 results in the Go To Case page 3200, as shown in Fig. 32. This allows the user to search for a case by entering the case-id, target id, and target type in the appropriate fields.

A variety of review, analysis and investigative actions can be taken as work on a 15 case proceeds. Internal actions are actions that the system completes when instructed to do so by an authorized user. Sending a message and changing the case status are examples of internal actions. External actions are actions whose completion requires work to be done outside the system. Interviewing recipients, requesting patient records, visiting provider offices and reaching a settlement with a provider are examples of 20 external actions.

Fig. 33 illustrates the action screen 3300 for a case, for taking a new action. This page allows the user to select an action from the ACTION drop-down 3302 at the top of the page. In a Spyder embodiment, the actions include:

- Update Case Status
- Update Case Category
- Update Case Detection Source
- Update Estimated Recovery Amount
- Update Actual Recovery Amount
- Attach Claim

- Detach Claim
- Link Case
- Unlink Case

As noted, in other applications, other actions may be applicable.

5 The bottom part of the screen (which changes, depending upon what the user selected for an action) allows the user to enter the required data, including descriptive comments to clarify the action.

The system records the date, time, User ID and other data for all internal actions. However, in order to track external actions users must enter information into the system.

10 For example, when a case results in savings (or is far enough along to project savings, recoveries etc.) authorized users can enter projected/actual savings data. This is the case with all external actions as well as "final outcome" information such as legal or disciplinary actions taken against providers, recipients, or other entities.

15 The user may also view the Case Log 3400 for a case, as shown in Fig. 34. This page allows users to view (and operate on using the context menu frame 508) the case log. This log lists all events related to the case. The "?"-bubble 3402 in the case status column 3404 provides a longer text description of the "O" case-status. This "bubble-help" facility is used throughout the WebStation for providing longer text descriptions for coded fields.

20 The user may also conduct general searches for cases, accessing a Search page 3500 via the Search tab 874. Fig. 35 illustrates the Search page. This pages more flexible search criteria than the Go To Case page. The WebStation executes a database search from the constructed query on the Database 140 and provides the resulting cases in a Case List, as illustrated in Fig. 36.

25 Case List 3600 is a list of cases that met the specified criteria from a search, and, like all data table pages has a Context Menu Frame 508 to allow further manipulation of the results. Clicking on the CaseID takes the user to the corresponding Case Summary page.

Assign Cases

Management users have the authority to assign cases to system users. They can do this via two different mechanisms. First, the Management user can select a specific Case Id and associate the case with a list of User Ids. Only the specified users can perform case management functions on the case (Management users have review access to all cases).

The queue mechanism is another way to perform an assignment, but it enables users to perform case management functions on all of the cases in the queue. When new cases enter a queue, all users assigned to the queue have access to all of the cases in the queue. In this way, the queue is a more general way to assign cases to users. Instead of dealing with singles cases, Management users can make assignments based upon categories of cases (embodied in the queue definitions).

The focus of case assignment is to control who has authorization to work (or view) cases. This control is placed in the hands of Management users (see User-Type).

Generate Management Reports

Management users have the ability to understand the current state of the fraud and abuse effort, analyze the effectiveness of the many components that make up the overall effort, and recognize opportunities for improvement. The Case Management module provides a suite of flexible reports to support these process tracking and analysis needs.

1) Current Case Load

- Current total number of cases in each different status
- Current cases by case-age
- Current cases by target-class
- Current cases by geographic region
- Current cases by system user (investigator)

2) Time-trend Case Load

- Same as Current Case Load, but with historical values plotted

3) User-Activity (one report per system user)

- General summary (User Identity data, current case load, historical case load, current cases by status, current cases by age bin, current cases by geo-region)
- Detailed summary (all of above, plus age by status matrix, time-trend case load, complete list of cases with Case Id, Target Class, Target Id, case-authorization start and end dates, case age, current case status)

5 4) Actual/Projected Savings

- Detailed case list, with basic case info, actual dollars saved, projected dollars saved, totals
- Actual/Projected dollars saved by Target Class
- Actual/Projected dollars saved by System User
- Actual/Projected dollars saved by Geographic Region

10 5) Geographical

- Case status by geographical region (target zip, target county, etc.)

15 6) Provider-Specific

Depending upon the target, different reports are provided. For example, the subset of cases where the target is a provider can be further analyzed by stated type and specialty, by practice size, practice location, provider clusters etc. Recipient cases can be broken out by recipient age/gender, recipient zip code, recipient cluster, etc.

20 **Case Management and Report Navigation**

If the user is “in the context of” a specific suspect (meaning they are viewing a report that relates to a specific individual suspect), and the user selects a case management tab from the Main Menu Frame 502 (Go To Case 880 or Assign Case 872), the WebStation 150 assumes the user wants to work with the case for that suspect. (If the user selects Go To Case 850 then the Case Summary 3100 for that suspect is shown on-screen; if the user selects Assign Case 872 the software assumes they want to assign a case for that suspect.)

Whenever the user is on a case management page (of any kind), they still see the Main Menu Frame 502 common to nearly all WebStation pages. The tabs in the Main Menu Frame 502 let the user go back to navigating reports—specifically the Home 820 and Search 874 tabs.

5 **Web CaseBook**

As a case is worked users perform different analyses, some using the WebStation 150, others taking advantage of external analysis tools and investigative processes. One mechanism for documenting a case is the Web CaseBook. As the case evolves, users can add reports generated within the Analysis WebStation to the Web CaseBook. The end 10 result is a record (in the form of a web site) that includes summary reports, reviewer/ investigator nates, case status updates and other information relating to the case.

Because detection models and rules are executed against data from a changing data warehouse, a thorough record of what the data looked like at the time of investigation is needed. The CaseBook meets this need.

15 Ideally, the Web CaseBook would have a hierarchical (outline) structure, but initially a single-level list of reports, comments and analyses will be provided. Each item in the list is stored as HTML and is linked to a title-entry in the main page of the CaseBook. At any point during the life of a case, the Web CaseBook can be viewed using a web browser. In a future version we will provide a CaseBook editor to help users 20 develop cleaner looking Web CaseBooks that include documents from external sources.

VIII. WebStation Configuration File

The WebStation is a general-purpose software application. It can be configured, using a partly automated and partly manual process, to meet the user-interface needs of any of a wide range of predictive solutions. The WebStation's flexibility is provided in 25 part by the WebStation Configuration File 190. The configuration file provides a way to generate a large portion of the WebStation application software. Specifically, the configuration file enables the WebStation Builder 195 process to generate code (software) that displays tabular data in reports within the Report Tree. The WebStation Configuration File 190 contains a description of which data in the System Database 140

is to be shown on-screen, in a tabular report, which reports can be linked to, how the reports are linked to (which data fields are links), how the data appears on-screen (color, font, etc.), the table layout (which fields are visible/hidden, order of fields), and a variety of other details. The high degree of configurability makes the WebStation deployment across a variety of predictive applications far more practical and efficient than custom development using a similar software specification (source code re-use).

The WebStation configuration file is used to define the default setup for the WebStation. This section describes exemplary configuration parameters for creating WebStation pages, where a page is a formatted view into the WebStation Database. Note that the WebStation configuration file can also define other global preferences, e.g. the default colors and font sizes, but those options are not discussed in this section. In a preferred embodiment, a WebStation configuration tool is used to configure pages for a WebStation, and enforces the syntax and configuration requirements of the WebStation pages as described below. Those of skill in the art will appreciate that pages for the WebStation may be created in any HTML or even text editor, and are not limited to creation in the configuration tool.

Preferred Syntax of a Configuration File

A configuration file is built on the concept of configuration “elements,” each of which may have an ID (if there are multiple such elements), and a list of associated data attributes. This section describes the configuration elements and their attributes in detail, including any default values; those of skill in the art will be able to readily extend or alter types of elements, their definitions, or their default values.

The format of these elements and attributes is:

Element [ElementID], Attribute1, Attribute2, ... ;

A list of attributes is always terminated with a semi-colon.

Attributes may be separated by a comma, as shown, but the comma is not necessary, and all white space is ignored.

If an element requires an ID, then it must be the first item following the element type.

Some attributes in the attribute list are just "flags," but many require a value, i.e., attribute=<value>.

5 If an attribute takes a list of values, then the list is enclosed in parentheses. Also, many attributes need not be specified explicitly because there is a default value for each (optional attributes will be shown in this section in square brackets).

The concept of a "parameter string" is used several times in this second. A parameter string is a string with embedded parameters that are expanded at run-time.

10 Parameters can be:

1. A parameter that is passed into a page when it is displayed. It is specified as

\$<param_name>

2. A global parameter (currently none are defined). These are specified as

\$<param_name>

15 3. A WebStation-defined columnID. Note that these can only be used in certain situations, i.e. when creating a SQL clause, or when passing column data from one page to another. These are specified as #<column_ID>.

An example of a parameter string is:

String = "The quick \$fox_color fox jumps over the lazy
20 \$animal_type."

In the above example, the string would be created with the current values for the parameters fox_color and animal_type. Substituting columns works a bit differently and will be explained later.

All ID's in the configuration file must be combination of alphanumeric or '_' characters, starting with either a '_' or a letter.

Comments may be interspersed throughout the configuration file, using either /* <comment> */ notation for large comment blocks, or // <comment> notation for single line comments.

The configuration file is not case sensitive, so, for example ID “model_id” and “MODEL_ID” are identical.

There is one “master” configuration file, but configuration elements can be separated into multiple files and included in the master configuration file, as INCLUDE 5 “<filename>”.

Pages

The primary configuration element in the WebStation configuration file is the “Page”. There are two types of pages that the WebStation works with:

“Table” pages—These are the most configurable and most common pages that the 10 WebStation displays. A table page consists mainly of a spreadsheet-style data display, with a page title and columns header. Setting up a table page mainly consists of defining the SQL and the resultant columns that are to be displayed.

“Form” pages – These are all other types of pages that can be a combination of graphs, charts and user input fields. Since the possibilities for such pages are endless.

15 Defining a form page comprises defining a page title, a list of parameters that the page requires, and either the names of the JavaScript methods that are used to build and take action on the pages, or the name of an HTML file.

A page is specified using the following configuration element:

Page <name>

20 Type = TABLE|FORM

[CopyOf = <existing_page>]

[Params = (param1, param2,...)]

- Every page must be given a valid, unique ID because pages are always referenced by name
- Type must be one of TABLE or FORM
- CopyOf is only available for TABLE pages and is an optional attribute that indicates that this page is a slight modification of the given existing_page.
A copied page assumes exactly the same column set as the existing_page,

but allows modification of the SQL where, order by, and having clauses. It also allows a different variation of hidden/shown columns as well as the column order of appearance.

- Params is an optional attribute that lists all the parameters required to build the page. Each parameter must be a valid ID. When a page is linked from another page, the link is checked to make sure that all the required page parameters are being passed. The WebStation configuration tool will generate an error if this condition is not met.

5

The following set of configuration elements follow a Page declaration because

10 they are used to define the contents of a page. Note that the order of appearance is not important, as long as all the elements belonging to a page occur after the Page declaration of that page, and before the Page declaration of the next page.

Title

15 The “Title” configuration element is used to specify the text in the top row of the table, above the column headers. It is specified as:

```
Title <title_param_string>;
```

Note, any HTML tags can be embedded in the string, to customize the display. For example, the following title string with put a line break (HTML tag
) in after the parameter model, and before the “for provider”:

20 Title “Clients from model ‘\$model
for provider \$provID’;

The following configuration elements are available only for table pages.

SQL

25 The “SQL” configuration element is used to define the clauses that are used to generate the SQL to build the report. Note that the column definitions occur with the column configuration elements (see the Column section, below) because columns have a lot of optional attributes and it is easier to keep all information related to a particular

column together. Also note that this is an optional configuration element for copied tables, but it is required for “original” tables.

SQL From=<from_param_string>
[Where=<where_param_string>]
[Having=<having_param_string>]
[OrderBy=(column1 [ASC|DESC], column2 [ASC|DESC], ...)]
[GroupBy=(column1, column2, ...)];

- “From” is typically just a string that indicates the name of the table or tables used in the query. It is possible, however, to specify a table using a page parameter. Do not use the word “from” in the string. Note that copied tables may not redefine the From clause.
- “Where” is an optional parameter string used to define the “where” clause for the query. The parameters used can either be standard parameters or the WebStation-defined column ID’s. Note that the use of column ID’s is optional when using simple column data because in most cases the column ID defined for the WebStation the same as the column name in the database. However, using a column ID can be useful when a SQL column is a more complicated calculation, like a combination of two or more other columns. Calculated columns, e.g. SUM(dollars), are preferably not used in the Where clause because those should be used in the Having clause. An example of a Where is as follows:

```
Where="#client_name like '%Smith%' AND  
service_date > '01-JAN-95' AND  
prov_id = '$provID'"
```

Note that operations that are AND’ed together are executed from the last operation to the first, so in the example, the where clause selects a given provider that matches the parameter provID. Then it looks for records where the values in table column service_date are after January 1, 1995. Finally, it looks for rows where the column ID client_name contains a sub-string of

“Smith” (note that client_name could be a concatenation of two table columns named first_name and last_name).

- “Having” is an optional parameter string that works almost exactly like the Where clause, except Having should only be used for calculated columns, i.e. sum, count, min, max, avg, etc. The reason for this is that Where acts on rows before they are retrieved, whereas Having is used to filter rows after they have been retrieved. Note that for efficiency, a Where clause should always be used for filters that act on a single row of data at a time.
- “OrderBy” is an optional list of columns used to sort the data. The first column listed is the primary sort key, the second column is the secondary sort key, etc. Ascending sort order is the default, but an ASC or DESC modifier can be used with each column to specify the sort order.
- “GroupBy” is an optional list of columns that defines how the data is to grouped, in the event that there is at least one calculated column requested, e.g. you want the total dollars overall (non-grouped) or the total dollars by client (grouped by client). The first column specifies the primary group, and the second column is the secondary group, etc. Note that there must be at least one calculated column (sum, count, etc.) in order for GroupBy to be valid, and if that is the case, then all the non-calculated columns should be specified in a GroupBy. If no GroupBy is specified and there is a calculated column, then the WebStation will group the non-calculated columns in their order of appearance.

Coloring

The “Coloring” configuration element is an optional field used to determine how the background colors of the data tables are alternated. This is a very useful parameter for allowing some degree of visual separation in the data.

Coloring ALWAYS|ONCHANGE

[Cols=(column1, column2, ...)]

- If “Coloring:” is defined, then it is followed by:
 - ALWAYS – Always alternate the background colors from row to row. This gives an appearance similar to a “green bar” paper printout report
 - ONCHANGE – Alternate the background colors based on the data in the table. Use the Cols attribute to specify the columns to monitor. This is very useful when the monitored columns match the OrderBy columns so that “blocks” of “like” data are colored together.
- Cols is used to specify the columns to be monitored for ONCHANGE coloring. It should not be specified for ALWAYS or NEVER Coloring. Any column ID may be used in the list, but this option is most useful if the columns match some subset of the columns specified in the OrderBy clause, if any.

Column

The “Column” configuration element is the most complicated page element in that it has the most possible attributes. It is used to define which database columns are chosen and how they are to be displayed. Note that redefining column attributes is not allowed for CopyOf pages

Column <columnID>
[SQL=<column_name or calculation>]
[DisplayName=<column header string>]
[SamplingTable=<database table for sampling>]
[DataWidth=<width of data>]
[Justify=LEFT|RIGHT|CENTER]
[Show/Hide]
[Type=DATE|TIME|STRING|INTEGER|FLOAT|MONEY|CATEGORY]
[Precision=<decimal precision for FLOATS>]
[Categories=(value1, value2, ...)]
[Info=<database table for info string>]

[Link=(<URL|pageID>,[<target_frame>],[parm1=<pValue1>,parm2=<pValue2>,...])]

- The columnID must be specified for every column. In most cases, the columnID for the column should be the same as the name of the column in the database, but it is not required to be so. A columnID must be unique for a given page.
- SQL is an optional attribute that specifies the SQL to build this column. Typically, this is just a simple column name, and if the columnID is the same as the requested column name, then the SQL attribute need not be specified at all. However, SQL must be specified for all calculated columns (sum, min, max, etc).
- DisplayName is an optional attribute that specifies the string that will appear in the column header of the data table. The string may contain embedded HTML tags if desired. If the DisplayName is not specified, it will use the columnID as the table column header.
- SampleTable is an optional attribute that is used to specify a database table that contains a list of all possible values for that column along with a random number for each column. This table can be used to sample the data displayed, and if it is specified a “Sample By” line will appear in the table Filter screen to allow the sample rate specification.
- DataWidth is an optional attribute that specifies the width of the data (in characters) that will appear in the table column on screen.
- Justify is an optional parameter to define how data is to be justified inside the column. If Justify is not specified, the column will be justified to the CENTER.
- Show/Hide are flag attributes that indicate whether or not the current column will be shown in the table or will be hidden from view. By default, all columns are shown, unless specified with a Hide flag.

- Type is an optional attribute that indicates the data type. This should match with the type of the column in the database, though a STRING type can always be used (though no always for the best results). By default, it assumes that the columns are STRING types.
- Precision is an optional attribute that is only allowed for FLOAT type data columns. It allows a limitation on the maximum number of digits after the decimal point that are allow in the display. By default, the data is displayed as it is stored in the database.
- Categories must be specified when the Type is CATEGORY. A CATEGORY type means that a given column is limited to a small number of elements (e.g. gender), and the Categories attribute is used to define all the possibilities. Usage of CATEGORY types only impacts the table Filter screen, where CATEGORY type columns are given filter options in “checkbox” format instead of the standard operations.
- Info is an optional attribute that is used to allow an English text description to be attached to a given coded value by specifying a database table that gives a code-to-English map. If Info is used, a small question mark “help” icon will appear next to all the rows in this column. Mouse-over on the icon will pop-up the English text.
- Link is an optional attribute that is used to define a hyper-link from this each cell in this column to another page. There are several arguments to the Link attribute:
 1. The first argument to Link is the page that is going to be linked to. Typically, this will be another page defined in the WebStation configuration file, specified with the pageID, but it can be a link to any desired URL by specifying the URL in a string.
 2. The second argument to Link is the target window that the linked page will be drawn in. This is an optional argument because, typically, WebStation pages are linked to the “dataFrame” window, which is the

main data display frame. However, any other target can be specified, including “`_parent`”, “`_self`”, “`_top`”, and “`_blank`”. Skip this argument to use the standard WebStation data window.

3. The rest of the arguments, if any, specify the parameters to be passed to the page that is being linked. Each parameter includes a “`parameter=<value>`” expression, where `parameter` is the parameter ID that the linked page is expecting, and `value` is the data that is being passed. Here, `value` is a type of parameter string and is typically one of:
 - a) A simple string constant, like the `pageID` of the current page, e.g. `currentPage=“claimsData”` would send a parameter called `currentPage` to the linked page with a value of “`claimsData`”.
 - b) A parameter. This is typically done to pass a current parameter to the next page. A common example is `modelID=$modelID`, which would send the current value of `modelID` to the linked page.
 - c) A `columnID`. This will pass the datum from the specified column in the current row to the linked page. Since the hyperlink is from a particular column in a particular row, it is often required that the linked page be passed additional information about the row that was selected. An example is `providerID=#provID`, where a parameter called `providerID` is passed the datum that resides in the `provID` column of the row that was clicked on.

Shown / Order

The “Shown” and “Order” configuration elements are only available for `CopyOf` pages. “Shown” is used to redefine only the Hide/Show attributes of the columns of the original page. “Order” is used to change the order in which the columns will appear. These attributes exist as the primary uses of the `CopyOf` concept, where there is a need for a page that runs the same query as an existing page, but a different set of columns is to be displayed and focused on, perhaps with a different sort order.

“Shown” is essentially a list of flags that indicate whether the indicated columns are shown (ON) or hidden (OFF). Any columns not specified in this list inherit the mode from the original page. “Shown” is an optional element and looks like the following:

Shown (columnID1 ON|OFF, columnID2 ON|OFF,...);

5 “Order” is a list of the columnID’s in the order that they will be displayed in the table. Note that all the columns from the original page must be represented in this list.

“Order” is an optional element and looks like the following:

Order = (columnID1, columnID2, ...);

Example Page Configuration

10 The following is an excerpt from an example configuration file for an example WebStation for healthcare fraud and abuse detection (e.g. a Spyder implementation). It gives the configuration for the main suspect list (page “modelScores”). That page has links to:

- 1) All the patients for each provider—page “provClients” linked from the “Clients” column
- 15 2) All the claim details for each provider—page “provClaims” linked from the “Details” column
- 3) All the reason reports

The provClients and provClaims reports are given, as well as a report called “patClaims” which is exactly like “provClaims” but it also takes a client ID as a parameter. Last, there is a report called “patClaimsPCG” which is also a copy of “provClaims” but takes both a client ID and a procedure group.

```
//  
25 // This is the page that displays suspect lists for models  
//  
PAGE       modelScores  
Type=TABLE  
Params= (modelID, navTree);
```

SQL From = "spy_mscr s, reas_lookup t1, reas_lookup t2,
reas_lookup t3, reas_lookup t4, reas_lookup t5"
Where = "s.reason_1 = t1.code AND s.reason_2 = t2.code AND
5 s.reason_3 = t3.code AND
s.reason_4 = t4.code AND s.reason_5 = t5.code AND MODEL_ID =
'\$modelID'"
OrderBy=(SCORE);

10 Title "Suspect list for model '\$modelID';

Column MODEL_ID
DisplayName="Model ID"
Hide;

15 Column SUSPECT_ID
DisplayName="Suspect ID"
Link=(summaryRpt, targID=#SUSPECT_ID,
targType=#SUSPECT_TYPE);

20 Column SUSPECT_TYPE
DisplayName="Suspect Type"
Hide;

25 Column SCORE
DisplayName="Score"
Type=INTEGER;

Column SCORE_DT
30 DisplayName="Score Date"
Type=DATE;

Column RANK
DisplayName="Rank"
35 Type=INTEGER;

Column TOT_PAID
DisplayName="Total Paid"
Type=MONEY
40 Justify=RIGHT;

```

Column      NUM_DETAILS
            DisplayName="Details"
            Link(provClaims, provID=#SUSPECT_ID)
            Type=INTEGER;

5          Column      NUM_CLNTS
            DisplayName="Clients"
            Link(provClients, provID=#SUSPECT_ID)
            Type=INTEGER;

10         Column      REASON_TXT
            DisplayName="Reasons"
            Justify=LEFT
            Grid= 5 x 1
            GridCol SQL="'(1) ' || s.reason_1 || ' - ' || "
            t1.description" Link(reasRpt, provID=#SUSPECT_ID,
            reason=#REASON_1)
            GridCol SQL="'(2) ' || s.reason_2 || ' - ' || "
            t2.description" Link(reasRpt, provID=#SUSPECT_ID,
            reason=#REASON_2)
            GridCol SQL="'(3) ' || s.reason_3 || ' - ' || "
            t3.description" Link(reasRpt, provID=#SUSPECT_ID,
            reason=#REASON_3)
            GridCol SQL="'(4) ' || s.reason_4 || ' - ' || "
            t4.description" Link(reasRpt, provID=#SUSPECT_ID,
            reason=#REASON_4)
            GridCol SQL="'(5) ' || s.reason_5 || ' - ' || "
            t5.description" Link(reasRpt, provID=#SUSPECT_ID,
            reason=#REASON_5);

15         Column      REASON_1
            DisplayName="Reason 1"
            Info="reas_lookup"
            Link(reasRpt, provID=#SUSPECT_ID, reason=#REASON_1)
            Hide;

20         Column      REASON_2
            DisplayName="Reason 2"

```

```

Info="reas_lookup"
Link(reasRpt, provID=#SUSPECT_ID, reason=#REASON_2)
Hide;

5   Column    REASON_3
      DisplayName="Reason 3"
      Info="reas_lookup"
      Link(reasRpt, provID=#SUSPECT_ID, reason=#REASON_3)
      Hide;

10
      Column    REASON_4
      DisplayName="Reason 4"
      Info="reas_lookup"
      Link(reasRpt, provID=#SUSPECT_ID, reason=#REASON_4)
      Hide;

15
      Column    REASON_5
      DisplayName="Reason 5"
      Info="reas_lookup"
      Link(reasRpt, provID=#SUSPECT_ID, reason=#REASON_5)
      Hide;

20
      // This page displays a provider's patients
      //
      PAGE      provClients
      Type=TABLE
      Params=(modelID, provID, navTree);

30
      Target    Param = provID
      Type     = "P";

      SQL      From     = "spy_mart"
      Where    = "$setID AND H1PPROV = '$provID'"
      OrderBy = (TOT_DOL DESC);

35
      Title     "Patient list for provider ID '$provID', used in model
      '$modelID' "

```

DATA DEFINITION

```

Column      H1PPROV
            DisplayName="Prov ID"
            Hide
            Link(provSum, provID=#H1PPROV);

5          Column      H1PCN
            DisplayName="Client ID"
            Link(clientSum, clientID=#H1PCN);

10         Column     PCN_AGE
            DisplayName="Age"
            Type=INTEGER;

15         Column     CLTGENDR
            DisplayName="Gender"
            Type=CATEGORY
            Categories=( "M", "F" );

20         Column     NUM_SRV_THIS
            SQL="COUNT(H1PCN)"
            DisplayName="# Services"
            Type=INTEGER
            Link(patClaims, clientID=#H1PCN, provID=#H1PPROV);

25         Column     AVG_DOL
            SQL="AVG(H1DPAID)"
            DisplayName="Average $"
            Type=MONEY
            Justify=RIGHT;

30         Column     TOT_DOL
            SQL="SUM(H1DPAID)"
            DisplayName="Total $"
            Type=MONEY
            Justify=RIGHT;

35         Column     PROC_GRP_NUM
            DisplayName="# Services, Proc Groups"
            Type=INTEGER
            Justify=CENTER;

```

Grid= 1 x 32

```
      GridCol SQL="SUM(PROC01)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC01")
5       GridCol SQL="SUM(PROC02)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC02")
       GridCol SQL="SUM(PROC03)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC03")
       GridCol SQL="SUM(PROC04)" Link(patClaimsPCG,
10  provID=#H1PPROV, clientID=#H1PCN, pcg="PROC04")
       GridCol SQL="SUM(PROC05)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC05")
       GridCol SQL="SUM(PROC06)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC06")
15  GridCol SQL="SUM(PROC07)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC07")
       GridCol SQL="SUM(PROC08)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC08")
       GridCol SQL="SUM(PROC09)" Link(patClaimsPCG,
20  provID=#H1PPROV, clientID=#H1PCN, pcg="PROC09")
       GridCol SQL="SUM(PROC10)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC10")
       GridCol SQL="SUM(PROC11)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC11")
25  GridCol SQL="SUM(PROC12)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC12")
       GridCol SQL="SUM(PROC13)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC13")
       GridCol SQL="SUM(PROC14)" Link(patClaimsPCG,
30  provID=#H1PPROV, clientID=#H1PCN, pcg="PROC14")
       GridCol SQL="SUM(PROC15)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC15")
       GridCol SQL="SUM(PROC16)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC16")
35  GridCol SQL="SUM(PROC17)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC17")
       GridCol SQL="SUM(PROC18)" Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC18")
       GridCol SQL="SUM(PROC19)" Link(patClaimsPCG,
40  provID=#H1PPROV, clientID=#H1PCN, pcg="PROC19")
```

```

        GridCol    SQL="SUM(PROC20)"  Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC20")
        GridCol    SQL="SUM(PROC21)"  Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC21")
5       GridCol    SQL="SUM(PROC22)"  Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC22")
        GridCol    SQL="SUM(PROC23)"  Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC23")
        GridCol    SQL="SUM(PROC24)"  Link(patClaimsPCG,
10      provID=#H1PPROV, clientID=#H1PCN, pcg="PROC24")
        GridCol    SQL="SUM(PROC25)"  Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC25")
        GridCol    SQL="SUM(PROC26)"  Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC26")
15      GridCol    SQL="SUM(PROC27)"  Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC27")
        GridCol    SQL="SUM(PROC28)"  Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC28")
        GridCol    SQL="SUM(PROC29)"  Link(patClaimsPCG,
20      provID=#H1PPROV, clientID=#H1PCN, pcg="PROC29")
        GridCol    SQL="SUM(PROC30)"  Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC30")
        GridCol    SQL="SUM(PROC31)"  Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC31")
25      GridCol    SQL="SUM(PROC32)"  Link(patClaimsPCG,
provID=#H1PPROV, clientID=#H1PCN, pcg="PROC32");

30      //
// This page display a providers's claims
//
PAGE      provClaims
          Type=TABLE
35      Params=(modelID, provID, navTree);

SQL        From="spy_mart"
          Where="$setID AND H1PPROV = '$provID'"
          OrderBy=(H1DFDOS1 ASC);

```

40

Target Param = provID
Type = "P";

Title "Claim data for provider '\$provID'. Data used in model
5 '\$modelID'"

Column H1PPROV
DisplayName="Prov ID"
Link(provSum, provID=#H1PPROV)
10 Hide;

Column H1PCN
DisplayName="Client ID"
Link(clientSum, clientID=#H1PCN);
15

Column H1ICN
DisplayName="Claim ID"
Link(claimSum, claimID=#H1ICN)
Hide;
20

Column H1DFDOS1
DisplayName="FDOS"
Type=DATE;

25 Column H1DPAID
DisplayName="\$ Paid"
Type=MONEY
Justify=RIGHT;

30 Column H1PROC
DisplayName="Procedure"
Info="proc_lookup";

35 Column H1DTOS
DisplayName="TOS"
Info="tos_lookup";

40 Column H1DPOS
DisplayName="POS"
Info="pos_lookup";

5

```

Column      H1DTCPT
DisplayName="Dx"
Info="dx_lookup";

10 // PAGE      patClaims
    Type=TABLE
    CopyOf=provClaims
    Params=(modelID, provID, clientID, navTree);

15 Target      Param = provID
    Type = "P";

SQL        Where="$setID AND H1PCN='$clientID' AND H1PPROV =
20 '$provID'";
Title       "Claim data for patient '$clientID', provider '$provID'.
Data used in model '$modelID'"

25 // PAGE      patClaimsPCG
    Type=TABLE
    CopyOf=provClaims
    Params=(modelID, provID, clientID, navTree, pcg);

30 Target      Param = provID
    Type = "P";

SQL        Where="$setID AND $pcg='1' AND H1PCN='$clientID' AND H1PPROV
35 = '$provID'";

```

Title "Claim data for patient '\$clientID', provider '\$provID'.
Data used in model '\$modelID'"

00000000000000000000000000000000